REMARKS/ARGUMENTS

Claims 1-3, 5-10, 12-17, and 19-24 are now pending in the present application. By the present action, claims 4, 11, and 18 are canceled; claims 1-3, 5-6, 8-10, 12-17, and 19-20 are amended; and claims 22-24 are added. Reconsideration of the claims is respectfully requested.

I. Objections to Specification

The specification stands objected to because pages 1 and 2 contain references to related applications whose serial numbers were not known at the time of filing. Paragraphs 1 and 2 have now been amended to show the serial numbers of these applications. No new matter has been added by these amendments to the specification. Therefore, this objection has been overcome.

Additionally, the title stands rejected as not being descriptive. In response, the title has been amended to be more descriptive. Therefore, this rejection is overcome.

II. Objection to Claims: Claim 5

Claim 5 stands objected to for a presumed typographical error, "routing" for "routine". The Examiner is thanked for finding this error. Applicants also discovered that corresponding claims 12 and 19 also contain the same typographical error. Each of these three claims has now been amended to correct the error. Therefore, this objection has been overcome.

III. 35 U.S.C. § 101: Claims 15-21

Claims 15-21 stand rejected under 35 U.S.C. § 101 as being directed towards non-statutory subject matter. This rejection is respectfully traversed.

Independent claim 15 has been amended to include the word "recordable-type" in the description of the computer-readable media. This amendment is believed to overcome the rejection.

IV. 35 U.S.C. § 102, Anticipation: Claims 1, 8 and 15

Claims 1, 8 and 15 stand rejected under 35 U.S.C. § 102(b) as being anticipated by **Smolders**, System for Tracing Hardware Counters Utilizing Programmed Performance Monitor to Generate Trace Interrupt After Each Branch Instruction or at The End of Each Code Basic Block, U.S. Patent No. 6,253,338, June 26, 2001 (hereinafter "**Smolders**"). This rejection is respectfully traversed.

The rejection states:

> As per claim 1 :
> Smolders teaches a method in a data processing system for monitoring the execution of a program, the method comprising:

> associating instructions for calls and returns in the program with a set of indicators (column 4, lines 15-18; column 3, lines 29-33; column 5, lines 20-24; column 5, lines 39-43) (Smolders associates all branches with indicators, this includes calls and returns. The indicators are T, the process information and counters.); and executing the program using a processor, wherein the set of indicators associated with the instructions causes the processor executing the instructions to generate data on calls and returns in the program (column 4, lines 18-21; column 4, lines 60-65).

Office Action dated January 12, 2006, page 3, item 7, emphasis added.

A prior art reference anticipates the claimed invention under 35 U.S.C. § 102 only if every element of a claimed invention is identically shown in that single reference, arranged as they are in the claims. *In re Bond*, 910 F.2d 831, 832, 15 U.S.P.Q.2d 1566, 1567 (Fed. Cir. 1990). All limitations of the claimed invention must be considered when determining patentability. *In re Lowry*, 32 F.3d 1579, 1582, 32 U.S.P.Q.2d 1031, 1034 (Fed. Cir. 1994).

Claim 1 and the other independent claims have been amended to more clearly recite that the method takes place in a system which contains a storage location associated with each instruction. This amendment finds support in the application on page 22, lines 9-27 and on page 25, line 18 through page 26, line 2 and provides clear distinctions over **Smolders.**

Claim 1, as amended, recites

> 1. A method in a data processing system for monitoring the execution of a program, the method comprising:
>
> in a system having an indicator location associated with each instruction, storing a respective indicator in the indicator location associated with each call and return in the program; and
>
> executing the program using a processor, wherein the respective indicators associated with the calls and returns cause the processor executing the instructions to generate data on calls and returns in the program.

In the invention recited in claim 1, the hardware is designed to provide assistance to debugging software by configuring an indicator location to be associated with each instruction. In this architecture, the system can be designed to check the indicator location each time an instruction is picked up for execution. If an indicator is stored in the indicator location, appropriate trace information can be generated. This means that counting of events can be enabled or disabled on an instruction-by-instruction basis.

In contrast, **Smolders** does not occur "*in a system having an indicator location associated with each instruction*" and thus is not able to store "*a respective indicator in the indicator location associated with each call and return in the program*". **Smolders** is not designed to provide the type of hardware support that is disclosed in the present application and so is unable to meet the features of this claim.

The rejection reads the indicators recited in claim 1 on "*T, the process information and counters*" shown in the following excerpts of **Smolders** (alleged indicators are underlined for convenience):

> As shown in FIG. 2, within the performance monitor 24 are monitor mode control registers (MMCR) 74 and 75, respectively, used for programming and one or more associated performance monitor hardware counters (PMC) 82 and 85 that used for counting operations.

Smolders, column 3, lines 29-33, emphasis added

> As described above, the instruction flow unit of the system processor unit 12 generates a trace interrupt after each branch instruction or at the end of each basic block of code as shown in step 30. By programming the monitor mode control register 74 to have the performance monitor counter 82 count instructions, the size of each basic block of code is determined.

Smolders, column 4, lines 15-21, emphasis added

> In the preferred embodiment, in order to be able to get a complete trace, i.e., not limited to a single process, the counter level tracing tool keeps track of the current process information (by way of example but not of limitation a process identifier (PID), thread identifier (TID), and program name.)

Smolders, column 4, lines 60-65, emphasis added

> Step 56 shows that if the current process is to be traced then a variable T is set equal to one wherein the next step 58 puts the process information, (for example, the PID, TID and program name) into a trace buffer and continues to step 60. Step 54 shows that if the current process is not to be traced, T is set equal to zero and continues to step 60.

Smolders, column 5, lines 19-26, emphasis added

> If it is then the current basic block address (i.e. tracing information) and the value of the hardware counters, 74 and 75, respectively, are placed in the trace buffer, as shown in steps 44 and 45, and the counter level tracing tool 31 continues to step 50.

Smolders, column 5, lines 39-43, emphasis added

In **Smolders**, *T* is a variable that is toggled to indicate whether or not the current process is to be traced. Examples of the process information are listed as "*a process identifier (PID), thread identifier (TID), and program name*" (see the third and fourth excerpts). Finally, counters 82 and 85 mentioned above will count operations, while counters 74 and 75 "*contain instructions for instructing their respective performance monitor counters, 82 and 85 respectively, to count a specified event ... [such as] counting the number of cycles during a selected executing process or the number of load/store misses occurring within an L2 cache*".

While *T*, the *process information*, and the *counters* do take part in the monitoring process, these "indicators" are not capable of providing an indicator location for each instruction. Consequently, **Smolders** does not meet the claim feature of "*in a system having an indicator location associated with each instruction, storing respective indicators in the indicator locations associated with calls and returns in the program*". Therefore, the rejection of claim 1 under 35 U.S.C. § 102(b) has been overcome.

Furthermore, **Smolders** does not teach, suggest, or give any incentive to make the needed changes to reach the presently claimed invention. Absent the examiner pointing out some teaching or incentive to implement **Smolders** with a location for each instruction, one of ordinary skill in the art

would not be led to modify **Smolders** to reach the present invention when the reference is examined as a whole. Absent some teaching, suggestion, or incentive to modify **Smolders** in this manner, the presently claimed invention can be reached only through an improper use of hindsight using the applicants' disclosure as a template to make the necessary changes to reach the claimed invention.

Since claims 8 and 15 are system and computer program equivalents to claim 1, the same distinctions between **Smolders** and the claimed invention in claim 1 will also apply for these claims. Consequently, it is respectfully urged that the rejection of claims 1, 8, and 15 have been overcome.

V. 35 U.S.C. § 103, Obviousness: Claims 2-7, 9-14, and 16-21

The examiner has rejected claims 2-7, 9-14, and 16-21 under 35 U.S.C. § 103(a) as being unpatentable over **Smolders** in view of **Subrahmanyam**, Transparent instrumentation for computer program behavior analysis, U.S. Patent No. 5,987,250, November 16, 1999 (hereinafter "**Subrahmanyam**"). This rejection is respectfully traversed.

All of the claims cited in this rejection are dependent on one of claims 1, 8, and 15 and thus inherit the allowability of their parent claims. Additionally, **Subrahmanyam** does not make up for the shortcomings of **Smolders**. This patent does not show "*a system having an indicator location associated with each instruction*", but discloses:

> The user also specifies locations in the application program object file which we will call probe locations, step 52 in FIG. 2. Probe locations are selected locations in the application program at which software probes will be used. However, the software probes themselves are not inserted into the application code at the probe locations. Rather, a flag or marker is provided that simply identifies the location as a probe location. The probes themselves typically are stored in a separate or appended file, step 54, for example in a probe source file 56.

Subrahmanyam, column 4, lines 9-17

Subrahmanyam uses a method different from Applicant's method for tracing, i.e., the insertion of a flag at locations that will be monitored and an external probe that will be invoked when a flag is detected. This is not the same as having a system configured in the recited manner. Additionally, **Subrahmanyam** uses only a software method of tracing a program, while **Smolders** provides a software method that is dependent on hardware assistance. One of ordinary skill in the art would not seek to combine these two references because of their disparate methods of providing a trace. Using the method of **Subrahmanyam** would involve ignoring the hardware assistance of **Smolders**, while using the method of **Smolders** would waste the additional software support provided by **Subrahmanyam**. These two systems are not compatible with each other because they rely on very different mechanisms to operate.

Neither does either of the references supply a motive to combine these references to meet the claimed invention. The rejection asserts that the combination would have been obvious since it "*allow for*

studying program behavior without affecting the program behavior" (Office action dated January 12, 2006, page 6, lines 1-2). However, **Smolders** already asserts that the method disclosed in this patent performs "*without introducing any overhead or modifying the code*" (**Smolders**, column 1, lines 64-67). Therefore, there is no incentive to combine **Smolders** with another program that simply accomplishes the same results in another manner.

Further, Applicants assert that a combination of the two patents would not reach the invention recited in these claims. As noted above, neither of the references relied on is run on a "*a system having an indicator location associated with each instruction*"; therefore, any combination of the two will not reach the claimed invention.

Therefore, the rejection of claims 2-7, 9-14, and 16-21 under 35 U.S.C. § 103(a) has been overcome.

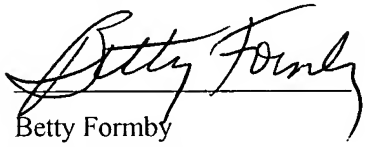
VI. Conclusion

It is respectfully urged that the subject application is patentable over **Subrahmanyam** and **Smolders** and is now in condition for allowance.

The examiner is invited to call the undersigned at the below-listed telephone number if in the opinion of the examiner such a telephone conference would expedite or aid the prosecution and examination of this application.

DATE: April 12, 2006

Respectfully submitted,



Betty Formby
Reg. No. 36,536
Yee & Associates, P.C.
P.O. Box 802333
Dallas, TX 75380
(972) 385-8777
Agent for Applicants